BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
Second Semester 2003-2004

Course Title : OPERATING SYSTEMS **Course No *CS C372 & IS C362***
Component : Test I (Regular) ***Open Book Component***
Weightage : 10% **Max Marks: 10** **Date : 05-03-2004**

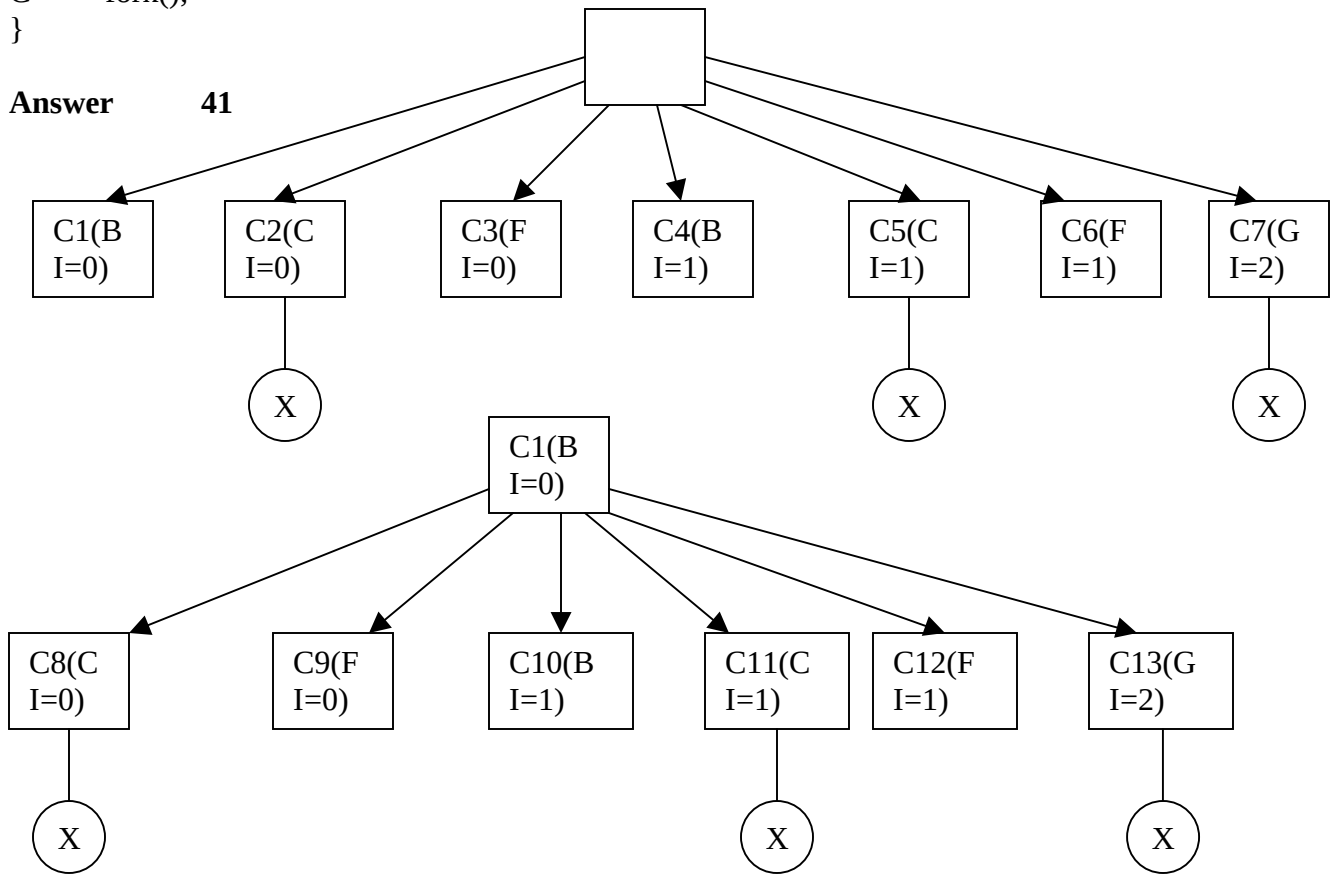
Note: Attempt all the Questions. Start each answer from a fresh page.

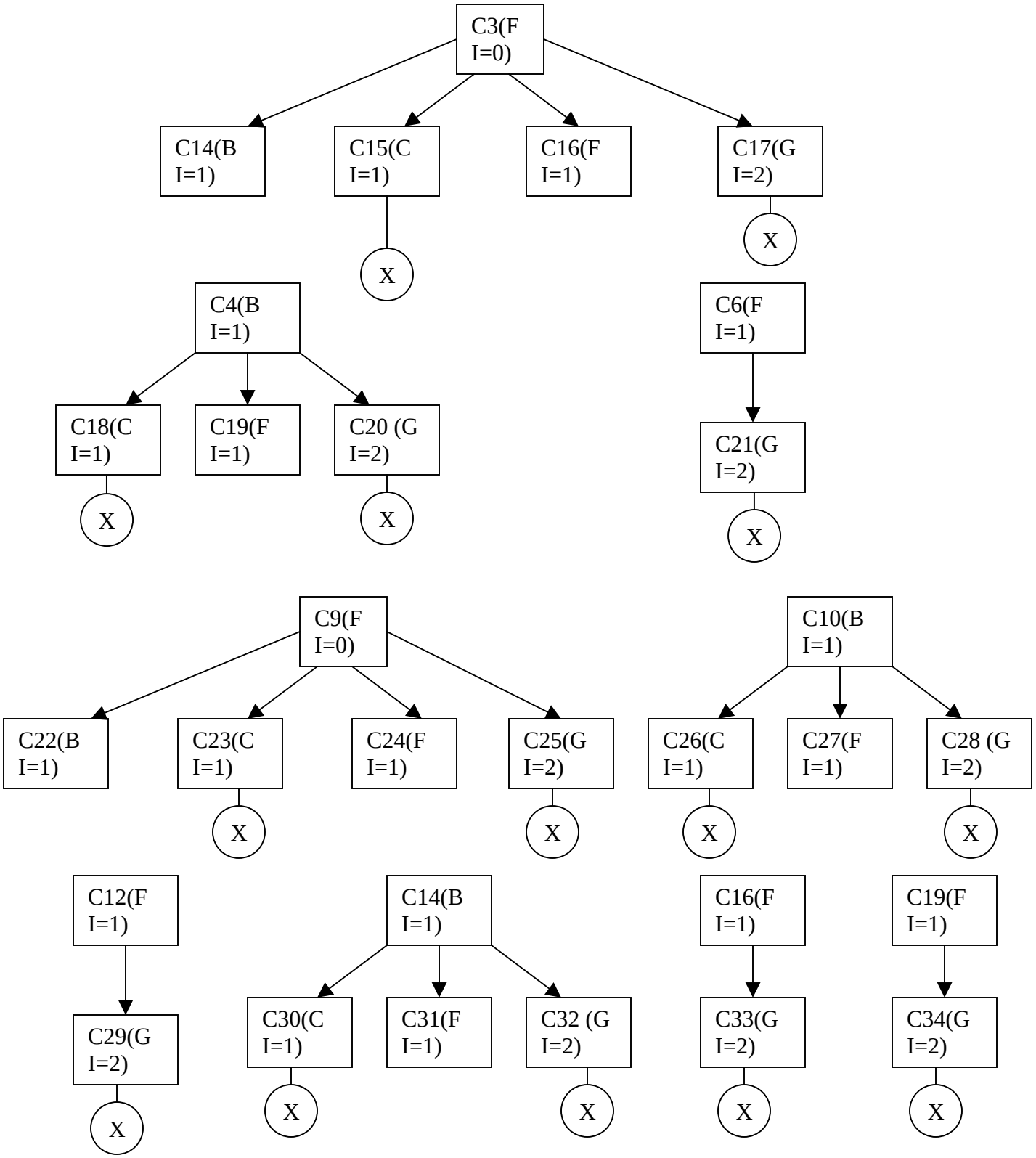
Question #1 **(2 Marks)**

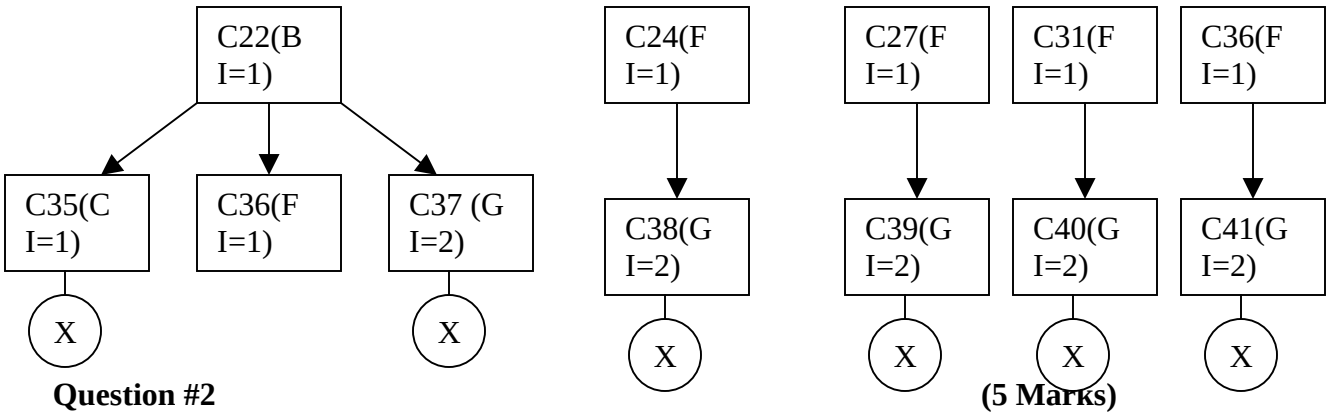
Find the number of new processes created.

```
void main()  {
        int i;
A       for(i=0;i<2;i++)  {
B               fork();
C               if(!fork())  {
D                       execl("/bin/ls","ls",NULL);
E                       fork();
                }
F               fork();
        }
G       fork();
}
```

Answer 41




C3(F I=0)
C14(B I=1)
C15(C I=1)
C16(F I=1)
C17(G I=2)
X
X
C4(B I=1)
C6(F I=1)
C18(C I=1)
C19(F I=1)
C20 (G I=2)
C21(G I=2)
X
X
X
C9(F I=0)
C10(B I=1)
C22(B I=1)
C23(C I=1)
C24(F I=1)
C25(G I=2)
C26(C I=1)
C27(F I=1)
C28 (G I=2)
X
X
X
X
C12(F I=1)
C14(B I=1)
C16(F I=1)
C19(F I=1)
C29(G I=2)
C30(C I=1)
C31(F I=1)
C32 (G I=2)
C33(G I=2)
C34(G I=2)
X
X
X
X



Question #2 **(5 Marks)**

A multilevel feedback queue algorithm works with the following condition

Number of queues 3 (Q1,Q2 and Q3)

Scheduling algorithm Q1 & Q2 uses R R, Q3 uses preemptive priority

Method used to upgrade a process No upgrading among queues (Priority value of processes in Q3 will increase by 1 for every 3 units of time waiting in Q3)

Method to demote a process Q1→After 2 units of time, Q2→ After 4 units of time (if preempted the process will continue remain in the front of the queue till the quantum time expiry), Q3→highest Priority process in the queue will get the first chance.

Method used to determine which queue a process will enter

P1, P3 & P5 are entering through Queue #1 (Q1)

P2 & P4 are entering through Queue #2 (Q2)

Process	T.CPU burst	CPU burst	I/O burst	Priority	Arrival time
P1	21	7	5	10	0
P2	18	9	2	6	1
P3	14	4	8	5	3
P4	19	8	4	7	6
P5	15	5	5	8	9

Calculate Average waiting time, average turn around time, and CPU utilization

P1	P2	P3	P2	P1	P5	P1	P3	P4	P5	P1	P3	P3	P5	P1	P5	P1	P3	P1
P5	P1	P3	P5*	P2	P3*	P2	P4	P1	P2	P4	P1	P4	P1	P2*	P1*	P4	IDLE	P4*
2	3	5	8	9	11	14	16	20	23	24	26	28	30	32	35	36	38	40
42	43	45	48	53	55	59	63	64	67	69	71	73	77	79	80	84	88	91

CPU utilization **87/91 = 95.6%**

Average Turn around time = (80+76+52+85+39)/5 = 334/5 = 66.9

Average Waiting time = (49+58+14+58+14)/5 = 193/5 = 38.6

Question #3 **(3 Marks)**

Consider the following test program for an implementation of join. When a parent thread calls join on a child thread the parent does the following

1. If the child is still running the parent blocks until the child finishes
2. If the child has finished the parent continues to execute without blocking.

```
int x=2;
void ThreadTest()
{
        thread t1,t2;
        t1 = new thread("A",1);
        t2 = new thread("C",1);
        t1->setPriority(10);
        t2->setPriority(15);
        x=x*5;
        t1->fork(A,0);
        x = x+10;
        t2->fork(C,0);
        x = x – 5;
        t1->join();
        t2->join();
}
```

```
void A(int arg)
{
        thread *t3;
        t3 = new thread("B",1);
        t3->setPriority(20);
        x = x * 2;
        t3->fork(B,0);
        x = x / 4 ;
        t3->join();
}
```

```
void B(int arg)
{
        x = x * x;
}
```

```
void C(int arg)
{
        thread *t4;
        t4 = new thread("A",1);
        t4->setPriority(10);
        x = x - 10;
        t4->fork(A,0);
        x = x + 5 ;
        t4->join();
}
```

(A) Assume that the scheduler run threads in Round Robin with no implicit time slicing (i.e. non-preemptive scheduling), priorities are ignored, and threads are placed on queues in FIFO order. What will be the value of x after execution?

(B) Now assume that the scheduler runs threads according to priority (the high priority value thread on the ready queue will run first and when a thread is added to the ready queue it will preempt the current thread if the new thread has higher priority If the priority of the main thread is 0 what will be the value of x after execution.

Answer

(A) 4

```
x=x*5          (M)
x=x+10
x=x-5

x=x*2          (A)
x=x/4

x=x-10         (C)
x=x+5

x=x*x          (B)

x=x*2          (A)
x=x/4

x=x*x          (B)
```

(B) 11020

```
x=x*5          (M)

x=x*2          (A)

x=x*x          (B)

x=x/4          (A)

x=x+10         (M)

x=x-10         (C)

x=x+5          (C)
```

x=x*2 (A)

x=x*x (B)

x=x/4 (A)

x=x-5 (M)